

CAS FEE 2020

Informationen zu
Projekten
Prüfung, Noten

Michael Gfeller mgfeller@hsr.ch

Silvan Gehrig sgehrig@hsr.ch

Markus Stolze mstolze@hsr.ch



HSR

HOCHSCHULE FÜR TECHNIK
RAPPERSWIL

FHO Fachhochschule Ostschweiz

PROJEKT INFORMATIONEN

■ Face2Face Unterstützung bei Projekten / Übungen

- Jeden Mittwoch um 16.00. Raum 1.213 / TBD / Web Konferenz
- Voranmeldung an mgfeller@hsr.ch
 - Bei grösseren Problemen

■ Fragen per E-Mail an mgfeller@hsr.ch oder sgehrig@hsr.ch

- Jederzeit

PROJEKT 1

Projekt 1: Notizen Verwalten

■ Ziel von Projekt 1

- Grundlagen festigen von CSS/JS/HTML

■ Aufgabe

- Erstellen einer Notizenapplikation

The screenshot shows the 'Note Pro' application interface. At the top, there's a browser-like address bar with 'http://www.notes.ch'. Below it, a 'Create new Note' button and a 'BlackWhite-Style' dropdown menu are visible. A navigation bar contains four buttons: 'By finish Date', 'By created Date', 'By Importance' (which is highlighted), and 'Show finished'. The main content area displays a list of notes categorized by date. Under 'Nächsten Mittwoch', there's a note 'CAS FEE Selbststudium / Projekt Aufgabe erledigen' with two lightning bolt icons. Under 'Heute', there's a note 'Einkaufen' with one lightning bolt icon, and a note 'Mami anrufen' with no icons. Each note has a text input field, a dropdown arrow, and an 'Edit' button. The 'Finished' checkbox is checked for the first two notes and unchecked for the third. The 'Irgendwann' category has a note '888 888 88 88' with no icons and an unchecked 'Finished' checkbox.

The screenshot shows the 'Note Pro' application interface in edit mode. At the top, there's a browser-like address bar with 'http://www.notes.ch'. Below it, a 'Title' input field and a 'Beschreibung' (Description) text area are visible. The 'Beschreibung' area contains the text 'HTML für die note App erstellen. CSS erstellen für die note App. [...]'. Below the description, there's a 'Wichtigkeit' (Importance) section with five lightning bolt icons, where the first two are filled and the next three are empty. Below that, there's an 'Erledigt bis:' (Due by) section with a date input field showing '//' and a calendar icon. At the bottom, there's a 'Speichern' (Save) button and a 'Cancel' button.

- **Startet heute.**
Aktion heute: Eintrag in Einschreibeliste (Name und GitHub URL)
<https://docs.google.com/spreadsheets/d/1ioKI-mxGm5fG4LNFvPehbV4wwjKUvQOGwMITQYI49A/edit#gid=0>
- **Bewertung: Pass/Fail (must pass)**
- **Arbeit alleine, aber öffentlich ([Github Account](#) in [Liste](#) eintragen)**
- **Abgabetermin 28.06.2020 – Mitternacht**
 - Branch erstellen mit dem Namen «Abgabe»
 - E-Mail erstellen mit folgendem Inhalt:
 - Subject: [CAS FEE] Abgabe {{Gruppennummer}}
 - Link zum Branch
 - Ein ReadMe auf GitHub, falls dieses notwendig ist.
- **Bei anderen Projekten nachzuschauen ist erlaubt, die Lösung muss aber eine Eigenleistung sein**
- **Erfolgreiche Abgabe von Projekt 1 ist Voraussetzung zum Start von Projekt 2**
 - Tipp: Bei «Problemen» früh melden und nicht bis zum Schluss warten

Projekt 1 – Einschränkungen

- Das Projekt sollte gut strukturiert sein aber weitgehend auf die Nutzung von Frameworks verzichten
 - Nutzung eines SPA Frameworks (Angular oder React+...) ist **nicht** erlaubt
 - Nutzung des MVC Patterns trotzdem sinnvoll
 - Nutzung von Layout / Styling-Framework wie Bootstrap nicht erlaubt
 - Flexbox und Grid nutzen
 - Nutzung von simpler Templating Engine (z.B. Handlebars) ist verlangt.
 - Nutzung von jQuery ist erlaubt aber nicht verlangt
- Ältere Browser müssen nicht unterstützt werden.

Projekt 1 – «Bewertung»

- **Das Projekt 1 wird in 5 Kategorien angeschaut. Jede Kategorie muss erfüllt sein.**
 - Funktionsumfang
 - Architektur
 - JS Qualität
 - CSS Qualität
 - HTML Qualität
- **Es wird ein generelles Feedback über alle Projekte geben.**
- **Bei größeren Problem in einem Projekt erhält man 2 Wochen Zeit um diese zu beheben.**

Projekt 1 – «Bewertung»

■ Funktionsumfang

- Der Funktionsumfang ist in den Wire-Frames dargestellt. Diese beinhalten u.a.
 - Anzeigen, editieren und erfassen von Notizen
 - Sortieren von Notizen
 - Filtern von „abgeschlossenen“ Notizen
 - Abspeichern der Daten auf dem Server
 - Wechseln des Styles

■ Allgemeine Code Guidelines

- Unterstützung modernen Browsers & Features (Ältere Browser können vernachlässigt werden)
 - CSS3+, HTML5+, ES6+
- Saubere Trennung von Struktur (HTML/HBS), Logik (JS) und Darstellung (CSS)
- Sauberen Code
 - DRY: Kein Copy-Paste-Code (auch keine 'ähnlicher Code'!)
 - Keine langen Methoden
 - Sprechende, konsistente Benennung von Variablen
- Kein CSS / JS im HTML
- Übersichtliche Projekt-Struktur

■ Architektur

- REST: Server und Client kommunizieren über JSON
- Client:
 - Server-Calls nur im Service Layer
 - Routing/Event-Handling nur im Controller
 - Rendering/DOM Manipulation nur in der View
 - View und Controller können im gleichem File definiert werden.
 - Kein HTML Zusammenbasteln aus String
 - Handlebars verwenden
- Server
 - Memory-Storage- / DB-Zugriffe nur im Service
 - Controller stellt Actions/Request-Handlers zur Verfügung
 - Router: Verknüpfung von Routen und Actions/Request-handlers

■ JavaScript

- Nutzung einer Template Engine auf dem Client (z.B. Handlebars)
- Keine Console Pollution
- Kein auskommentierter Code
- Kein «global namespace pollution»

■ CSS

- FlexBox / Grid
 - keine Float-, Table- oder Inline Layouts
 - keine unnötigen Klassen & ID's (Elemente über Struktur/Name selektieren)
- Inline Styles & Inline-Style-Klassen sind nicht erlaubt

■ HTML

- Korrekter Einsatz von Semantischem HTML
- Sinnvolle HTML Validation

Projekt 1 – Ablauf

Woche	Aufgabe-Grob	Details
1	Projekt auf Github anlegen und in Liste eintragen. Codewars-Account erstellen. HTML Gerüst erstellen für die WireFrames und Beginn CSS.	Ändern der Wireframes ist erlaubt.
2	https://www.codewars.com Rang 6 erreichen	
3	HTML & CSS mit Grid / Flex-layout	Einen zweiten sinnvollen Style ausdenken.
4	Hauptseite ausprogrammieren: Anzeigen der Einträge / Filtern / Sortieren Handlebars verwenden für das Rendern der Einträge. Detailseite ausprogrammieren: Erfassen / Editieren Style Switcher implementieren. Implementation als Single Page oder mit LocalStorage	Flex-layout im Projekt einsetzen.
5	JavaScript optimieren. Patterns anwenden. Nutzen von Klassen für die Datenhaltung.	(Revealing) Module Pattern für die "Datenklassen" erstellen. IIFE anwenden.
6	Client Modularisierung fortführen.	
7	Die REST API vom Server implementieren. Diese im Client anbinden.	Bonus: Neue Einträge sollen auf andern Browser sichtbar werden. z.B. durch Polling.
8	Angular Tutorial	
9	Finalisieren & Abgabe	

CODEWARS

← → ↺

www.codewars.com/dashboard

☆ JB 📷 ⋮

6 kyu136

Your Next Challenge...

JavaScript

Rank Up

TRAIN

SKIP

Upgrade To Codewars Red!

Support Codewars and get some fancy upgrades like Pro Stats, Head-to-head comparisons, No ads, Streaming Output a

LEARN ABOUT RED

Allies

You are automatically given an allegiance with anyone who is in the same clan as you. You can also become allies with other warrior or inviting new warriors to join.

Position	User	Clan
1	mstolze	Hochschule für Technik Rapperswil

5 kyu

Perimeter of squares in a rectangle

The drawing shows 6 squares the sides of which have a length of 1, 1, 2, to see that the sum of the perimeters of these squares is : $4 * (1 + 1 + 8) = 4 * 20 = 80$

Could you give the sum of the perimeters of all the squares in a rectangle n + 1 squares disposed in the same manner as in the drawing:

ALGORITHMS

MATHEMATICS

LOGIC

NUMBERS

Master

4 dan

3 dan

2 dan

1 dan

Proficient

1 kyu

2 kyu

3 kyu

4 kyu

Competent

5 kyu

6 kyu

Novice

7 kyu

8 kyu

Beginner

Kata

In our dojo, kata are real code challenges focused on improving skill and technique. Some train programming fundamentals, while others focus on complex problem solving. Each kata is crafted for and by the community.

Kyu/Dan Ranks

Each kata on the site is set to a Kyu/Dan rank, based on its subject area and difficulty. The community collectively determines rank in the Beta Process. Every new Codewarrior (user) on Codewars starts being ranked as 8 Kyu.

What are Kyu and Dan? The terms are borrowed from a system in Japanese martial arts, which is in turn borrowed from the game of Go. Kyu indicates the number of degrees away from master level (Dan). This is why they count downward. Once you reach master level, we count upward.

Currently Dan level content is reserved for later. You will only find Kyu level content to train on for now.

Earning Ranks

You can advance through the ranks by completing kata at or above your rank - the harder the kata the faster you advance.

HSR

HOCHSCHULE FÜR TECHNIK

RAPPERSWIL

FHO Fachhochschule Ostschweiz

- Anmelden mit Github-Account oder HSR oder eigener E-Mail
 - Nachteil HSR: geht verloren nach Abschluss
- Falls gewünscht Clan Eintragen
 - CASFEE2020
- Account Info Abgabe im Projekt 1 Info Sheet eintragen
- Zum Start von Projekt 2 muss Level 7kyu erreicht sein
- Zum Abschluss von Projekt 2 muss Level 6kyu erreicht sein
- Unregelmässig: Leaderboard & Erfahrungsaustausch

PROJEKT 2

Projekt 2: Single Page Applikation

- **Erfolgreiche Abgabe von Projekt 1 ist Voraussetzung zum Start von Projekt 2**
- **Erreichen von CodeWars Level 7kyu ist Voraussetzung zum Start von Projekt 2**
- **Projekt 2 wird in 2er Gruppen bearbeitet.**
 - 3er Gruppe ist möglich,
 - 1er Gruppe nur in Ausnahmefällen
- **Ziel von Projekt 2**
 - Vertiefen von Advanced Themen
 - Erstellen einer Single Page Anwendung unter Nutzung eines aktuellen SPA Frameworks (Angular oder React+Redux)

Projekt 2: Single Page Applikation

■ Freie Projektwahl mit Mindestanforderungen:

- User-Management
- Server mit Daten (REST-Routes)
- Die Themenwahl muss von einem Betreuer bestätigt werden
- Umsetzung entweder mit Angular oder React+Redux

■ Beispiel-Themen:

- Reddit (Einfach)
- Webshop
- Buchhaltungssystem
- Online-Schach (Schwer)

■ Ideen, Themen und Gruppenfindung ab: Freitag 06.07 (Ganztag)

- Repos anderer Mitglieder anschauen: Stil, Skill-Level,
- Diskutieren:
 - Zieltechnologien, Projekt-Ideen
 - Geplante Herausforderung: Halbmarathon, Marathon, Iron-Man, ...

■ Präsentation: 13. Januar 2021

■ Abgabe: 17. Januar 2020 (Sonntag nach Präsentation/Prüfung)

■ Weitere Informationen am Montag 06.07

PRÜFUNG & NOTEN

■ Zugelassen zur Prüfung ist:

- Wer 80% oder mehr des Unterrichts besucht hat.
- Wer CodeWars Level 6kyu erreicht hat.
- Wer Projekt 1 erfolgreich abgegeben hat (Bewertung “pass”).

■ Prüfung gilt nur als bestanden, wenn:

- Projekt 2 im Team erfolgreich abgegeben wurde.

- **Datum: 13. Januar 2021**
- **Start 15:30**
- **Closed Book, keine Computer**
- **1 Stunde**
- **Besteht aus Multiple Choice und «1 Satz Antwort-Fragen»**
 - Fragen, welche man als Front-End-Engineer in einem Bewertungsgespräch beantworten können sollte.
 - Fokus auf Themen die im Grundlagenteil (bis Sommerpause) behandelt wurden

Notenverteilung

Prüfung	40%
Projekt 2	60%

- **Studienreglement und Prüfungsrichtlinien im Moodle verfügbar**